|  |
| --- |
| **BỘ GIÁO DỤC VÀ ĐÀO TẠO**  **TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**  **D:\_Data_2019_2020\logo_utehy.png**  **TIỂU LUẬN MÔN LẬP TRÌNH PYTHON NÂNG CAO**  **Đề tài**  **Phân tích dữ liệu nhân sự cho công ty TNHH ABC**  NGÀNH: CÔNG NGHỆ THÔNG TIN  HỌC VIÊN: **NGUYỄN VĂN DƯƠNG**  **VƯƠNG THỊ THƯƠNG**    LỚP: H01222  HƯỚNG DẪN: **TS. NGUYỄN VĂN QUYẾT**  **HƯNG YÊN - 2023** |

# MỤC LỤC

Contents

[MỤC LỤC 1](#_Toc137654887)

[LỜI NÓI ĐẦU 2](#_Toc137654888)

[MÔ TẢ BÀI TOÁN 3](#_Toc137654889)

[GIẢI QUYẾT BÀI TOÁN 4](#_Toc137654890)

[1. Tính toán tỷ lệ giới tính công nhân viên của công ty ABC 4](#_Toc137654891)

[2. Phân tích cơ câu giới tính nhân viên trong mỗi phòng ban của công ty ABC 6](#_Toc137654892)

[3. Phân tích tỷ lệ tuổi lao động trong công nhân viên công ty ABC 8](#_Toc137654893)

[4. Phân tích cơ cấu nhân sự trong từng phòng ban của công ty ABC. 10](#_Toc137654894)

[5. Phân tích số lượng nhân sự được tuyển theo từng năm ở từng địa phương. 12](#_Toc137654895)

[6. Phân tích cơ cấu vùng miền trong nhân sự công ty ABC. 14](#_Toc137654896)

# LỜI NÓI ĐẦU

Trong thời đại ngày nay, tin học đã trở nên phổ biến với mọi người từ mọi cấp học, từ mọi ngạch-bậc của xã hội. Cùng với sự phát triển đó, mọi phần mềm ứng dụng tương ứng với từng chức năng cụ thể cũng đã ra đời. Song, không thể có được một phần mềm có thể cung cấp hết các chức năng cho mọi công việc, từ công việc kế toán, quản lý nhân viên đến quản lý nhân sự, quản lý bán hàng... Bởi, mỗi chức năng cần có một yêu cầu riêng, có những nét đặc trưng mà không phần mềm nào có thể đáp ứng được.

Do vậy trong việc phát triển phần mềm, sự đòi hỏi không chỉ là chính xác, xử lý được nhiều nghiệp vụ mà còn phải đáp ứng các yêu cầu khác như: giao diện thân thiện, sử dụng tiện lợi, bảo mật cao, tốc độ xử lý nhanh….

Ví dụ như việc quản lý nhân sự trong công ty TNHH ABC. Nếu có sự hỗ trợ của tin học thì việc quản lý nhân viên, tuyển dụng và đào tạo nhân viên sẽ thuận tiện, nhanh chóng và hiệu quả hơn. Ngược lại các công việc này đòi hỏi rất nhiều thời gian và công sức mà sự chính xác và hiệu quả không cao, vì hầu hết đều làm bằng thủ công trên excel khá vất vả.

Quản lý nhân viên, quản lý tuyển dụng và đào tạo là một công việc tương đối phức tạp, tốn nhiều thời gian và công sức. Chính vì vậy, tin học hóa trong lĩnh vực quản nhân sự, tuyển dụng, đào tạo là một yêu cầu tất yếu. Muốn quản lý tốt cần có được các phần mềm tốt, phần mềm phải đảm bảo được độ bảo mật cao, dễ sử dụng và nhiều tiện ích. Trong việc này, python được áp dụng khá nhiều vì tính chất đơn giản, mềm dẻo nhưng lại vô cùng hiệu quả, nhanh chóng và mạnh mẽ. Giúp cho việc phân tích các dữ liệu trở lên đơn giản và hiệu quả.

# MÔ TẢ BÀI TOÁN

Bài toán áp dụng cho phòng nhân sự công ty TNHH ABC.

Phòng nhân sự mong muốn phân tích dữ liệu nhân sự được tổng hợp từ khi thành lập công ty cho đến nay để phục vụ các nghiệp vụ tuyển dụng và đào tạo nguồn nhân lực. Dựa vào các kết quả của phân tích dữ liệu để đưa ra chiến lược nhân sự cho thời gian tiếp theo.

Dữ liệu đầu vào :

File csv tổng hợp thông tin nhân viên từ khi thành lập công ty (2006) cho đến nay (2023).

Thông tin nhân viên bao gồm :

* Mã nhân viên.
* Tên nhân viên.
* Quê quán
* Tuổi
* Phòng ban
* Chức vụ
* Trạng thái làm việc / nghỉ việc. Thời gian vào làm/ thời gian nghỉ.

Các yêu cầu phân tích bao gồm:

* Phân tích tỷ lệ giới tính.
* Phân tích tỷ lệ giới tính các phòng ban của công ty.
* Phân tích tỷ lệ tuổi của công nhân viên trong toàn công ty.
* Phân tích cơ cấu nhân sự trong mỗi phòng ban.
* Phân tích số lượng nhân sự được tuyển theo từng khu vực.
* Phân tích cơ cấu nhân sự theo vùng miền.

# GIẢI QUYẾT BÀI TOÁN

## Tính toán tỷ lệ giới tính công nhân viên của công ty ABC

import pandas  as pd

import numpy as np

import matplotlib.pyplot as plt

data = pd.read\_csv('DLNS-ALL.csv')

# phân tích tỷ lệ giới tính toàn công ty

# khởi tạo dữ liệu phục vụ cho việc liệt kê

sex\_names = np.array(['Nam','Nữ'])

sex\_counts = np.array([0,0])

# lọc danh sách

emp\_data = data.loc[1:, ['Status','Giới tính']]

item\_count = emp\_data.shape[0]

for index in range(0, item\_count):

    status = emp\_data.iloc[index]['Status']

    if status != 'Normal':

        continue

    IsMale = emp\_data.iloc[index]['Giới tính'] == 'M'

    if IsMale :

        sex\_counts[0] += 1

    else:

        sex\_counts[1] += 1

plt.title('Tỷ lệ giới tính trong công ty ABC')

plt.pie(sex\_counts, labels= sex\_names, startangle= 0, autopct='%1.2f%%')

plt.axis('equal')

plt.legend(bbox\_to\_anchor=(1, 0, 0.5, 1))

plt.savefig('a1.png')

plt.show()

A blue and orange pie chart

Description automatically generated with medium confidence

## Phân tích cơ câu giới tính nhân viên trong mỗi phòng ban của công ty ABC

Ý nghĩa :

Trong thực tế, mỗi phòng ban của công ty được bố trí tại một khu vực khác nhau. Tại mỗi khu vực đó cần tính toán tỷ lệ giới tính công nhân viên để tính toán cơ sở hạ tầng phục vụ cho đặc điểm giới tính ở mỗi khu vực riêng. Ví dụ để tính toán quy mô nhà vệ sinh, khu nghỉ ngơi…

import pandas  as pd

import numpy as np

import matplotlib.pyplot as plt

data = pd.read\_csv('DLNS-ALL.csv')

# phân tích tỷ lệ giới tính từng phòng ban

# lấy cột dữ liệu phòng ban và liệt kê các phòng ban

departments\_df = data.loc[1:, ['Department Name']]

department\_names = []

for index in range(0, departments\_df.shape[0]):

    name = departments\_df.iloc[index]['Department Name']

    if name not in department\_names:

        department\_names.append(name)

departments\_cnt = len(department\_names)

male\_counts = np.empty(departments\_cnt)

female\_counts = np.empty(departments\_cnt)

male\_counts.fill(0)

female\_counts.fill(0)

# chọn dữ liệu và lọc danh sách

employee\_df = data.loc[1:, ['Department Name','Status','Giới tính']]

item\_count = employee\_df.shape[0]

for index in range(0, item\_count):

    status = employee\_df.iloc[index]['Status']

    if status != 'Normal':

        continue

    IsMale = employee\_df.iloc[index]['Giới tính'] == 'M'

    department\_name = employee\_df.iloc[index]['Department Name']

    department\_idx = department\_names.index(department\_name)

    if IsMale :

        male\_counts[department\_idx] += 1

    else:

        female\_counts[department\_idx] += 1

# vẽ đồ thị

columns = np.arange(len(department\_names))

plt.bar(columns, male\_counts, label='Nam', color='red')

plt.bar(columns, female\_counts, label='Nữ', color='blue', bottom=male\_counts)

plt.title('Cơ cấu giới tính của các phòng ban trong công ty ABC')

plt.xlabel('Phòng ban')

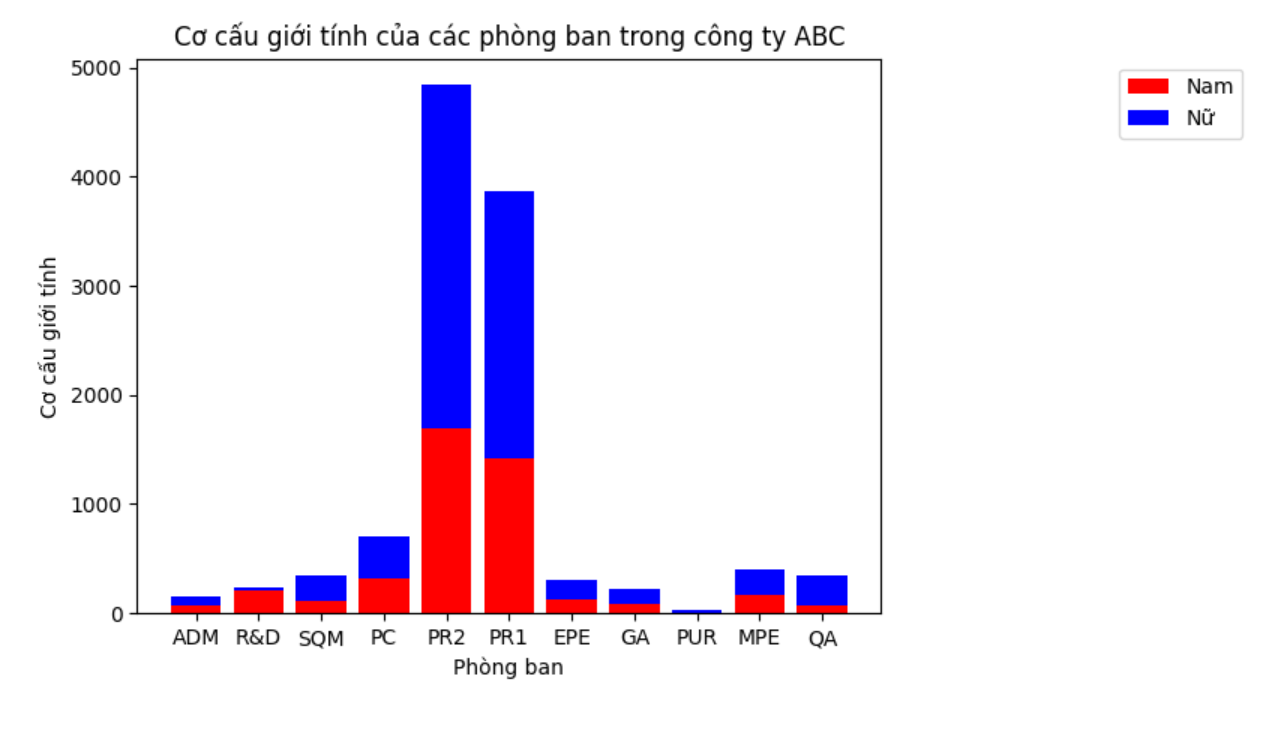
plt.ylabel('Cơ cấu giới tính')

plt.legend(loc='best',bbox\_to\_anchor=(1, 0, 0.5, 1),labels=['Nam','Nữ'])

plt.xticks(columns, department\_names)

plt.savefig('a2.png')

plt.show()



## Phân tích tỷ lệ tuổi lao động trong công nhân viên công ty ABC

import pandas  as pd

import numpy as np

import matplotlib.pyplot as plt

data = pd.read\_csv('DLNS-ALL.csv')

# phân tích tỷ lệ tuổi

age\_start = 18

age\_end = 60

age\_titles = np.arange(start= age\_start, stop = age\_end + 1, step= 1)

age\_counts = np.empty(age\_titles.shape[0])

age\_counts.fill(0)

# lọc danh sách

employee\_df = data.loc[1:, ['Status','Tuổi']]

for index in range(0, employee\_df.shape[0]):

    status = employee\_df.iloc[index]['Status']

    if status != 'Normal':

        continue

    age = employee\_df.iloc[index]['Tuổi']

    if(age < age\_start) or (age > age\_end):

        continue

    age\_index = int(age - age\_start)

    age\_counts[age\_index] += 1

plt.title('Tỷ lệ tuổi nhân viên trong công ty ABC')

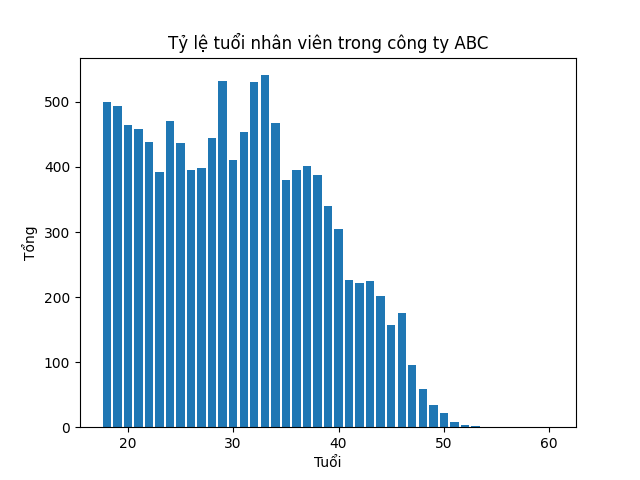
plt.xlabel('Tuổi')

plt.ylabel('Tổng')

plt.bar(age\_titles, age\_counts)

plt.savefig('a3.png')

plt.show()



## Phân tích cơ cấu nhân sự trong từng phòng ban của công ty ABC.

Ý nghĩa :

Phân tích tỷ lệ chức vụ trong các phòng ban của công ty. Thông qua đó đánh giá cơ cấu nhân sự xem có thiếu hay thừa nhân lực hay không.

# phân tích cơ cấu nhân sự

# lấy cột dữ liệu phòng ban và liệt kê các phòng ban

departments\_df = data.loc[1:, ['Department Name']]

department\_names = []

for index in range(0, departments\_df.shape[0]):

    name = departments\_df.iloc[index]['Department Name']

    if name not in department\_names:

        department\_names.append(name)

departments\_cnt = len(department\_names)

# chuẩn bị dữ liệu cho phân tích

pos\_names = ['Operator','Assistant Leader','Leader','Senior Leader','Staff','Technician','Assistant Supervisor','Supervisor','Junior Management','Middle Management', 'Senior Management']

pos\_cnt = len(pos\_names)

counters = []

for pos\_name in pos\_names:

    row\_data = np.empty(departments\_cnt)

    row\_data.fill(0)

    counters.append(row\_data)

# lấy các cột phòng ban, trạng thái, vị trí

# sau đó phân tích dữ liệu thống kê

employee\_df = data.loc[1:, ['Department Name','Status','Position Type']]

item\_count = employee\_df.shape[0]

xxx = 0

for index in range(0, item\_count):

    status = employee\_df.iloc[index]['Status']

    if status != 'Normal':

        continue

    department\_name = employee\_df.iloc[index]['Department Name']

    pos\_name = employee\_df.iloc[index]['Position Type']

    department\_idx = department\_names.index(department\_name)

    pos\_idx = pos\_names.index(pos\_name)

    counters[pos\_idx][department\_idx] += 1

# vẽ đồ thị

columns = np.arange(len(department\_names))

bottom = counters[0]

counter\_cnt = len(counters)

for index  in range(0, counter\_cnt):

    if index == 0:

        plt.bar(columns, counters[index], label=pos\_names[index])

        bottom = counters[index]

    else:

        plt.bar(columns, counters[index], label=pos\_names[index], bottom=bottom)

        bottom = bottom + counters[index]

plt.title('Cơ cấu nhân sự các phòng ban trong công ty ABC')

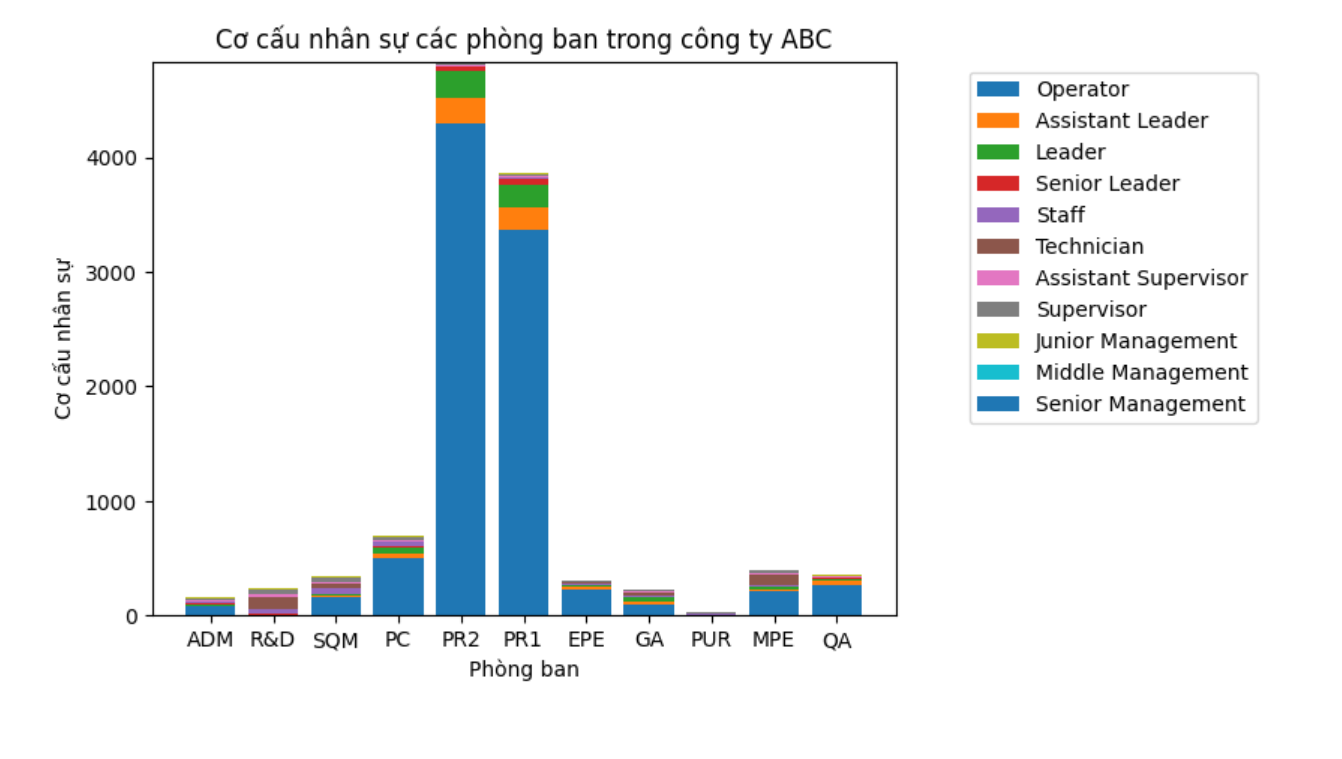
plt.xlabel('Phòng ban')

plt.ylabel('Cơ cấu nhân sự')

plt.legend(loc='best',bbox\_to\_anchor=(1, 0, 0.5, 1), labels = pos\_names)

plt.xticks(columns, department\_names)

plt.show()



## Phân tích số lượng nhân sự được tuyển theo từng năm ở từng địa phương.

Ý nghĩa :

Có thể dùng để phân tích tiềm năng ứng viên theo từng vùng miền. Phục vụ cho nhu cầu tuyển dụng trong tương lai

import pandas  as pd

import numpy as np

import matplotlib.pyplot as plt

data = pd.read\_csv('DLNS-ALL.csv')

# phân tích cơ cấu nhân sự

# lấy cột dữ liệu phòng ban và liệt kê các phòng ban

departments\_df = data.loc[1:, ['Department Name']]

department\_names = []

for index in range(0, departments\_df.shape[0]):

    name = departments\_df.iloc[index]['Department Name']

    if name not in department\_names:

        department\_names.append(name)

departments\_cnt = len(department\_names)

# phân tích số lượng nhân sự tuyển dụng qua các năm ở từng địa phương

# lấy cột địa phương

province\_df = data.loc[1:, ['Tỉnh']]

provinces = []

for index in range(0, province\_df.shape[0]):

    province = province\_df.iloc[index]['Tỉnh']

    if province not in provinces:

        provinces.append(province)

years = []

for year in range(2005,2024):

    years.append(int(year))

counters = []

for province in provinces:

    counter = np.empty(len(years))

    counter.fill(0)

    counters.append(counter)

# lấy dữ liệu cần thiết

employee\_df = data.loc[1:, ['Join date','Tỉnh']]

for index in range(0, employee\_df.shape[0]):

    province = data.iloc[index]['Tỉnh']

    province\_idx = provinces.index(province)

    joindate = data.iloc[index]['Join date']

    year = int(joindate[-4:])

    year\_idx = years.index(year)

    counters[province\_idx][year\_idx] += 1

plt.figure(figsize=(16,10))

plt.xticks(years)

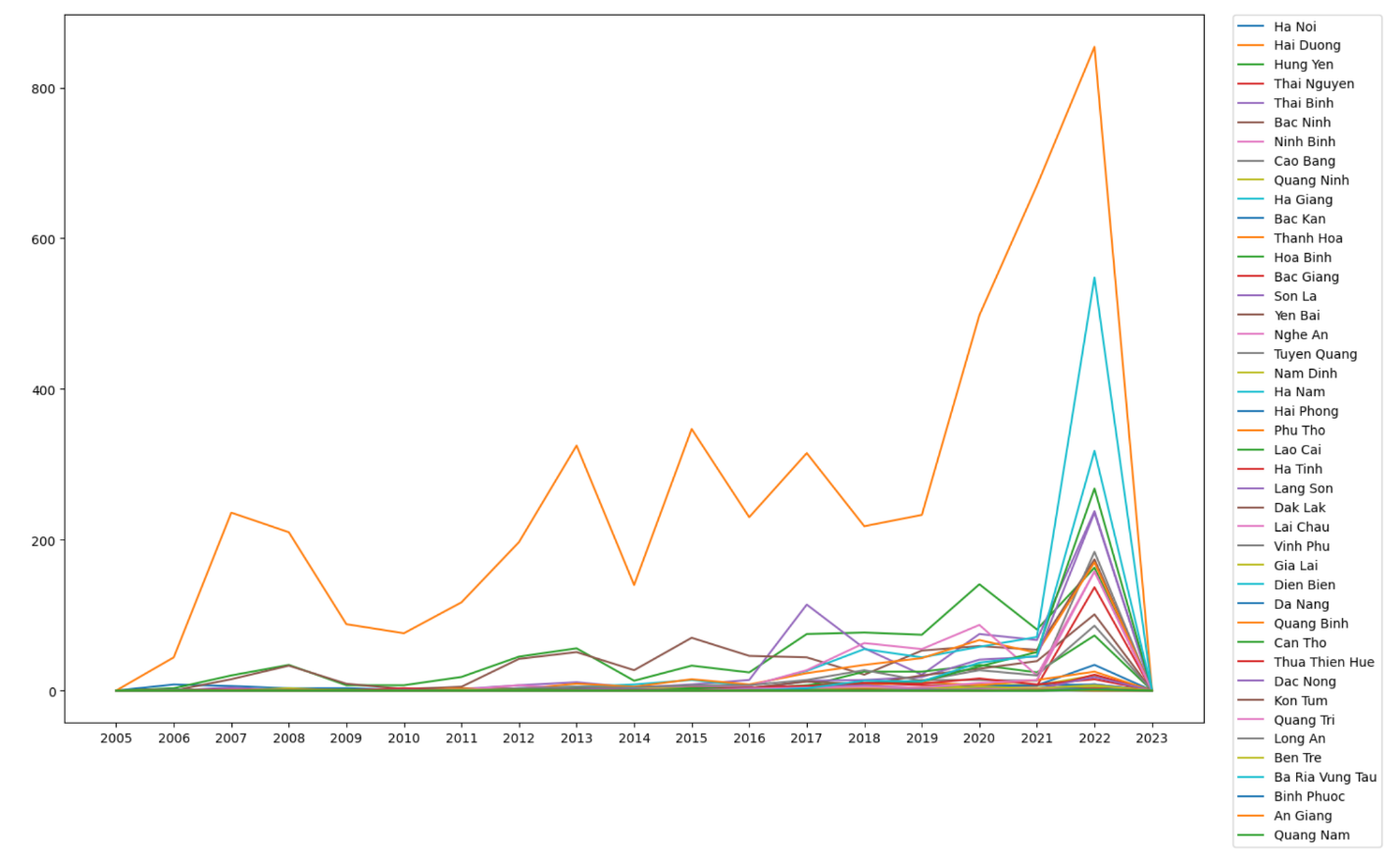
for index in range(0, len(counters)):

    plt.plot(years, counters[index], label = provinces[index])

plt.xticks(years)

plt.legend(loc='best',bbox\_to\_anchor=(1, 0, 0.5, 1), labels = provinces)

plt.show()



## Phân tích cơ cấu vùng miền trong nhân sự công ty ABC.

Ý nghĩa : Phân tích được cơ cấu vùng miền để có những chính sách phù hợp với nhân viên. Ví dụ tổ chức các ngày lễ của vùng miền đặc trưng cho nhân viên.

import pandas  as pd

import numpy as np

import matplotlib.pyplot as plt

data = pd.read\_csv('DLNS-ALL.csv')

# phân tích số lượng nhân sự tuyển dụng qua các năm ở từng địa phương

provinces = ['Hai Duong','Hung Yen','Ha Noi','Ha Giang','Bac Ninh','Thai Binh','Dien Bien','Khac']

counters = np.empty(len(provinces))

counters.fill(0)

# lấy dữ liệu cần thiết

employee\_df = data.loc[1:, ['Tỉnh']]

for index in range(0, employee\_df.shape[0]):

    province = data.iloc[index]['Tỉnh']

    if province in provinces:

        counter\_idx = provinces.index(province)

        counters[counter\_idx] += 1

    else:

        counter\_idx = provinces.index('Khac')

        counters[counter\_idx] += 1

print(counters)

plt.title('Tỷ lệ vùng miền của nhân sự trong công ty ABC')

plt.pie(counters, labels= provinces, startangle= 0, autopct='%1.2f%%')

plt.axis('equal')

plt.legend(bbox\_to\_anchor=(1, 0, 0.5, 1))

plt.show()

